

The Automated Satellite Data Processing System

Image Processing

The Automated Satellite Data Processing System: Image Processing

Published 28 March 2008

Table of Contents

I. APS Image Processing Library 1

II. Command Line Reference 2

List of Figures

1. Output Image 6

List of Examples

1. Adding a bathymetry product to a file	5
2. Adding a depth product to a file	5
3. Creating a TIFF true color image	26
4. Extracting Remote Sensing Reflectance Products into a single ENVI file	30
5. Extracting Diffuse Attenuation into a GeoTIFF file	30
6. Applying Low Pass Filter to All Products in a File	32
7. Applying Center-Weighted Edge Detection to only Reflectance Products in a File	32
8. Producing a Difference Between Files Created Using Two Versions of APS	35
9. Producing a Difference Between Files Created Using Two Versions of APS For Only Reflectance Products	35
10. Producing a Difference Between Files Created Using Two Versions of APS For Only Reflectance Products (Output Same as an Input)	36
11. Determine Difference Results on a Series of Input Files	36
12. Warping Reflectance Data Using Specified Map	49
13. Warping All Data Using Specified Map	49
14.	54
15. Producing a Water Mass Classification Product from Selected Input Data	57
16. Extracting Data From a Product File	59
17. Computing Statistics For a Product	65

Part I. APS Image Processing Library

The chapters in Part I form a User's Guide for the image processing programs available within the Automated Processing System.

Part II. Command Line Reference

The chapters in Part II form a reference guide for each program available in the Automated Processing System for image processing.

Name

imgBathy -- create a bathymetry product

imgBathy
imgBathy [*options*] *ifile* [*ofile*]

Description

This program is used to create a bathymetry product for the given input file. The resulting bathymetry will be appended to *ifile* written as a 2-D float32 array and will be named “bathymetry”. If the user appends a second file name (*ofile*) the “bathymetry” data set will be written to that file using the first one for navigation only.

By default, the input bathymetry file is \$APS_DATA/ETOPO2v2.DOS file.

Options

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-f bathymetry_file

This option is used to specify the input bathymetry file.

-n name

This option is used to specify another name for the output data set. The default is bathymetry

-o name=name,format=format,conv=conv

Define output file.

keyword	description
name	name of the output file
format	format of the output file
conv	convention of the output file

--help

Print out a small help guide.

--version

Print out version of software and quit.

Environment Variables

APS_DATA

The directory where all the data files exist. Defaults to \$APS_DIR/data.

Files

BATHY.DAT

The bathymetry file. It is the ETOP5 2-minute gridded product created by NGDC. Each value is in whole meters. The file used by APS originated with SeaDAS.

Examples

In this example, the bathymetry is appended to the given level-3 data file.

Example 1. Adding a bathymetry product to a file

```
$ imgBathy S2000065175121.L3_HNAV_MSB
```

Now, suppose that instead we want to put the bathymetry in a second file called GOM_BATHY.hdf and we want to call the array depth instead of bathymetry.

Example 2. Adding a depth product to a file

```
$ imgBathy -n depth S2000065175121.N3_HNAV_GOM GOM_BATHY.hdf
```

Name

imgBrowse -- create a quick-look image

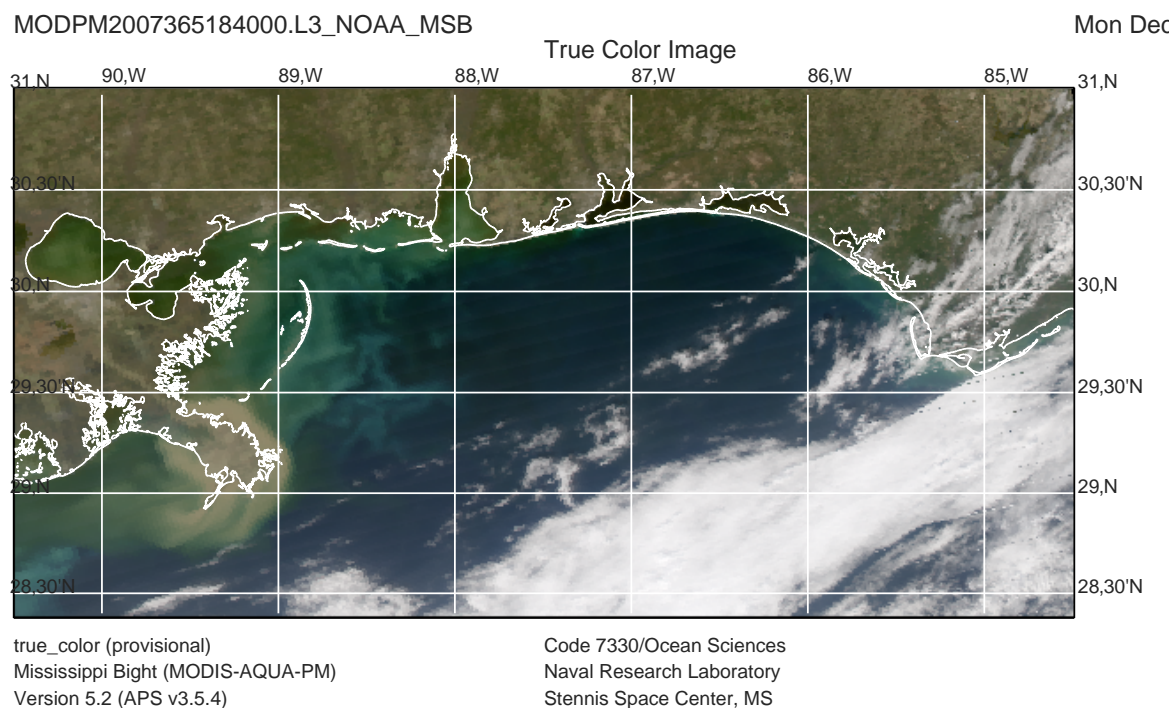
imgBrowse

imgBrowse [*options*] *ifile product ofile*

Description

This program creates a reduced image of the given product in a standard graphic format. The exact formats available depend on the software build process as it uses external libraries. Use the *--help* option to see which formats are known.

Figure 1. Output Image



See the **EXAMPLES** section below for many different calling ideas. Also see the **CAVEATS** section below for current problems.

The input file is normally an APS file that contains certain attributes and structures required for proper execution of imgBrowse program. The following list shows which attributes are used and any alternatives. See [data_users_guide.pdf](#) for greater detail.

These attributes are used to annotate the image to describe the type of data that is being displayed and how it was processed.

Attribute	Description
fileTitle	title of file
timeStartYear	Start Year of data
timeStartDay	Start Day of year of data
timeStartTime	Start Time in milliseconds of year of data
timeStart	ASCII start time string (see ctime(3))
timeEndYear	End Year of data
timeEndDay	End Day of year of data
timeEndTime	End Time in milliseconds of year of data
timeEnd	ASCII end time string (see ctime(3))
compType	composite type of data
compMaxPixel	composite maximum pixel (see imgMean(1))
inputMaxPixel	alias for compMaxPixel
sensor	sensor name
Sensor Name	alias for sensor
sensorPlatform	sensor platform
processedVersion	processing version of data
inputMasks	name of masks used in data processing
inputMasksInt	bit-mask of masks used in data processing
mapProjection	name of map projection

These attributes are used to determine how to extract and create the image product.

Attribute	Description
scalingSlope	calibration scale for conversion of input data to geophysical values
slope	
scale_factor	
Slope	
scalingIntercept	calibration offset for conversion of input data to geophysical values
intercept	
add_offset	
Intercept	
XXXSlope	calibration scale for conversion of input data to geophysical values in XXX units. For example, FahrenheitSlope
XXXIntercept	calibration offset for conversion of input data to geophysical values in XXX units
browseFunc	default scaling function for imgBrowse
browseRange	default data ranges for imgBrowse
validRange	data ranges for imgBrowse if above missing
browseSlope	browse scale for conversion of geophysical values to image
browseIntercept	browse offset for conversion of geophysical values to image
browseCT	default imgBrowse color table number

These attributes describe the MET/OZONE files used by the ocean color programs for MODIS/SeaWiFS processing. If found and the user has select option *-J*, then the MET wind data will automatically be overlayed as vectors and the OZONE data will automatically be overlayed as contours (with no associated colorbars).

Attribute	Description
met1	First input MET file
met2	Second input MET file
met3	Third input MET file
ozone1	First input OZONE file
ozone2	Second input OZONE file
ozone3	Third input OZONE file

Options

-l

Builds the image at 1-to-1 resolution; do not try to resize the image to a reasonable size. A built-in limit of 4096x4096 should not be exceeded. This option can not be used with *-s* option (it will override). If *irp/irl* are used with *-B* option, this option should be set.

-a fontfactor

Used to increase the size of the fonts by multiplying the font size calculations (for FreeType rendered fonts or Encapsulated PostScript output only). For example, a value of 1.2 will make all fonts 20% larger than normal. The factor is applied equally to all text. The *fontfactor* value must be greater than zero. THIS SHOULD BE THE FIRST OPTION SPECIFIED.

-A fontpath

Defines the font path for True Type fonts or the font face for PostScript (e.g., *Helvetica* (default), *Times*, etc.) If the file \$APS_DATA/imgBrowse.font exists and True Type fonts are being used, then this file is used. Otherwise, the compiled in values of are used:

-b filename

Reads in the given blotch file and overlays each region over the image.

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

Defines a subsection of the original image for output. Not to be used with the -s option.

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

Note that by default imgBrowse will try to resize the image to fit a computer screen, so that if the user sets irp/irl the results might not be as expected. The -l option will force imgBrowse to use the defined sizes.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Defines a subsection of the original image based on geographical coordinates for output. Cannot be used with the -s option.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-c options

Sets options that control the colorbar, where:

keyword	description
num_labels	the number of labels (> 1)
label_format	C-style printf format
ct_num	the colortable to use
lut	file containing a colortable
location	where to place the color ('bottom' for bottom, 'left' for left of image)
reverse	reverse the colortable
brighten	add/subtract bias to/from colortable
bad	use to set the bad data value color (default is white)
invalid	use to set the no data value color (default is black)
background	use to set the background color (default is white)

The keywords *reverse* and *brighten* must *follow* the color table entry (either *ct_num* or *lut*).

The current available colortables are

num	description
0	NRL Rainbow
1	NASA Bio Sphere
2	NASA NDVI
3	Blue to Red
4	Negative Radiance
5	Goddard Chlorophyll-a
6	Visibility
7	Sea Surface Temperature
8	Johns Hopkins Univ. APL SST
9	Hue 2
10	Rainbow
11	Purple Red Strips
12	NRL Visibility
13	EOS colortable
99	Greyscale

The *lut* table refers to a colortable in either the SeaDAS format (*.lut*), the NSIPS format (*.ct*), or XVision format (*.pal*).

-C num

Sets colortable (see table above).

-C product=name,range=n:m:l,scale=n.m,offset=n.m,netcdf=file

Sets the contour overlay options, where:

keyword	description
netcdf	NetCDF file containing data to produce contours for. Must contain 1 or 2-dimensional latitude and longitude arrays describing the navigation at every point.
hdf	APS file containing data to produce contours for. Must contain APS compatible navigation.
product	name of product to produce contours for
timestep	0 for first timestep, 1 for second, ... etc. (most significant dimension of 3 or 4 dimensional product)
depth	0 for first depth, 1 for second, ... etc. (second most significant dimension of 4 dimensional product)
range	3 element, colon separated declaration of contour points. the first element is the number of contour points, and the second and third elements are the minimum and maximum contour points respectively. The contour points will be evenly spaced between
	the minimum and maximum.
contour_point	4 element, colon separated declaration of an individual

The color table used for contour drawing is set in the usual way (i.e. by using the `-C` option with a number, or by setting the color table options). The color table should be set before defining the contour parameters. Different color tables may be used for each contour by choosing a color table between each contour statement. The default color table may be reset by setting the color table to `-999` (i.e. `-C -999`).

`-C range=21:-.5:.5,units=meters,product=Surface_Elevation,netcdf=file`

This will read the product "Surface_Elevation" from the file "file" and create contours at 21 levels spaced evenly between `-.5` and `.5`. The latitude and longitude arrays will be read from their default names of Latitude and Longitude respectively.

`-d`

Turns on debug output (may be very verbose)

`-D date`

Sets given string as the date field (upper right corner). Otherwise date from input file is written here.

`-e title`

Sets the title string (upper left corner) to *title* (defaults to input file name).

`-E`

Adds text indicating this product is "EXPERIMENTAL". Additionally if there is a file named `experimental.png` in `$APS_DATA`, it will use that file as the watermark (`-w` overrides this behavior).

`-f [linear|log|log10]`

Selects Function used for scaling (defaults to *linear*).

`-F name`

Sets name of the data set containing the flags. Defaults to `"l2_flags"`.

`-g draw=[0|1],color=R:G:B,spacing=f,lat_spacing=lat,lon_spacing=lon lat_label=[1|2|3],lon_label=[1|2|3],lat_label_color=R:G:B, lon_`

Sets grid overlay options, where:

keyword	description
draw	turn on (1 - default) or off (0) grid lines
color	color (0.0 - 1.0) used to draw grid lines
spacing	lat/lon spacing between grid lines
lat_spacing	spacing between latitudinal grid lines
lon_spacing	spacing between longitudinal lines
lat_label	1 - for labels on left of image 2 - for labels on right of image 3 - for labels on both sides of image
lon_label	1 - for labels on left of image 2 - for labels on right of image 3 - for labels on both sides of image
lat_label_color	set color of latitude labels
lon_label_color	set color of longitude labels
lat_label_modulo	skip n grid lines for latitude labels
lon_label_modulo	skip n grid lines for longitude labels
line_width	set the width of the grid lines (default=1.5)
style	line style: 0-solid, 1-dashed, 2-dotted
label_format	format of labels (dd-decimal degrees) (dms-degrees, minutes, seconds)

For example, **-g color=0.0:1.0:0.0,spacing=0.5** draws green gridlines spaced every 0.5 degrees."

-G dir

Name of the input data directory. Defaults to \$APS_DATA.

-i

Create the image only. In this case the border, the color table, and annotations, etc. will not be drawn. However, the logo and watermark are still drawn.

-I interlace=[0|1|2],band=0,flip=[0|1]

Sets options for the input image, where:

keyword	description
interlace	the interlace mode for the 3-D input data set
band	selected band to display
flip	flag to flip the input image (0-no/1-yes)

-j file

Sets the bathymetry option for overlaying a 30m bathymetry contour over image using the given input file. See imgBathy(1) and imgMakeLatLon(1) on how to create this file. The function **apsMakeBathy** in the **apsScripts.sh** file automatically creates the appropriate file.

-J

Used to overlay the met/ozone data used by the file over the image. The MET wind data is overlayed as vectors and the OZONE data is overlayed as contours.

-k

Defines the location (if any) of the classification of the image. May be set to 0 (no classification) or 1 (classification written to top and bottom on image).

-K options

This option is used to create a colorbar consisting of breakpoints. That is data between breakpoints will be all set to the same color. Each break point is separated by semicolons (;) and the color is set after the break point with a colon (:). Each component of the color (red,green,blue) are separated by commas (.). For example,

-K 0:0,0,0;20:1,0,0;40:0,1,0 -r 0,40 ifile sst sst.jpg

will create an image of sst such that all pixels less than 0 will be black. Those from 0 to 20 will be red, those from 20 to 40 will be green.

-l

Sets the options used to control the landmask, where:

keyword	description
draw	draws (1) or does not (0) the land mask
color	defines the color of the landmask
file	sets the name of the input landmask file

The input landmask file may be the default SEADAS file `$APS_DATA/landmask.dat` or an NSIPS created landmask file.

-L file

Sets the logo file

-L file=file,width=w,height=h

Sets the options used to control the logo, where:

keyword	description
file	sets the name of the input logo file
xsize width w	sets the width of the displayed logo
ysize height h	sets the height of the displayed logo
x	sets x-location of the displayed logo
y	sets y-location of the displayed logo

-m mapFile:mapName

Use the following mapFile and mapName for navigation.

-M NAME1;NAME2;NAME3

Name of masks to use. Multiple mask names must be separated by a semi-colon ";". A color may be assigned for each mask by following the mask name with a colon ":" and an RGB triplet separated by commas. For example, **-M CLDICE;ATMFAIL**

or

-M CLDICE:0.8,0.8,0.8;ATMFAIL:0.5,0.5,0.5

-N file

Read in file and place these notes at the bottom on the image. This file is a simple UNIX text file and each line is written verbatim to the bottom on the image.

-o format=format,...

Sets options that control the output file, where:

keyword	description
format	output format of file (e.g. jpg)
compress	TIFF file compression ('j' for JPEG, 'd' for deflate)
quality	JPEG compression quality (1-100, defaults to 85)
level	deflate compression level (1-9, defaults to 6)
alpha	write alpha channel (1) or not (0)

The format is a typical file extension, like jpg or png.

An alpha channel will only be written to a format that can handle one. Presently, this option is limited to TIFF and PNG formats.

-O x

Sets the offset in the log10 function.

-p file

Place points or symbols from the file on the image (see FILES below).

-P

Get political boundaries from \$APS_DATA/polbnd.dat in Piskor's format and overlay them on the image.

-Q file

Creates a thumbnail file in JPEG format. This thumbnail is a small version of the output image without all the annotation and border.

-r min,max

Select range of input data for scaling. Defaults to min/max of each product's validRange parameters. Failing that it uses the min/max of product. Failing that it is set to min/max of all reals.

-s w,h

Sets the size of output image. Used to reduce or enlarge the size of the output. Defaults to size of original image. Cannot be used with the -B or -l options.

-S slope,intp

Set the slope and intercept to use for scaling the image

-t file=file,color=R:G:B,line_width=w,font_size=f,label=label,skip=n,delim=d,cols=lat:lon

Set options that provide ship track overlays, where:

keyword	description
file	pathname of the ship track file
delim	set the delimiter between columns in file. Default is tab.
cols	set columns which contain the lat/lon (0-relative)
skip	number of records from top of file to skip
color	set desired color for ship track file
line_width	set the width of the ship track (default=1.0)
font_size	set size of label
label	set ship track label (default=filename)
no_label	not draw label

These options allow the user to apply a ship track to the output image. The default track file is a UNIX tab-delimited text file with latitudes in column 1 and longitudes in column 2. The latitudes and longitudes are expected to be in decimal degrees in the range (-90 to 90) and (-180,180), respectively.

If the input file uses a different delimiter, the user can change it using the **delim** keyword. The first **n** lines might be skipped (if for example there is a header), using the **skip** keyword. Lastly, if the latitudes and longitudes are located in different columns, then the user may set them using **cols**, like '**cols=2,3**'.

The other options allow the user to define the color, line width, and labeling for the ship track.

-T type

Type of sensor, for example, 'SeaWiFS', 'AVHRR', etc.

-u units

sets the units for the given product.

-U type

Used to create a world file that describes the navigation of the output file. A world file is a simple text file that will contain either map coordinates (type=1) or geographical coordinates (type=2). The world file will be named after the input file with the addition of the letter **w** (e.g. if the output file is `world.jpg`, the world file will be `world.jpgw`).

-V nogaps=file,met=file,pctides=file,timestep=[0|1], color=R:G:B,factor=n.m,scale=n.m,units=string, arrow_px_len=n,line_width=w

Vector overlay option

keyword	description
nogaps	file containing nogaps wind vectors to overlay
met	file containing NCEP MET wind vectors to overlay
pctides	file containing PCTides current vectors to overlay
ncom	file containing NCOM current vectors to overlay
vfile	for the v-component NCOM vector file.
timestep	0 for first nogaps timestep, 1 for second
depth	0 for first depth, 1 for second, ... etc.
factor	define the representation of one arrow length
units	string representing specific units
scale	scale to modify the input vector data
offset	offset to modify the input vector data
subsamp	subsampling ratio for input data
latitude	sets latitude
longitude	product name sets longitude
lonlat	product name switch notion of lonlat ordering
ucomp	sets U-component product name
vcomp	sets V-component product name
thresh	minium windspeed threshold needed to draw vector
color	set desired color (0.0-1.0) for

arrow_px_len	vector file 22 length of standard vector (default is 20.0 pixels)
--------------	---

For example, **-V factor=0.25,scale=0.5144,units="m/s",pctides='file',color=0.5144)**, in green, where a vector of 20 pixels represents 0.25 meters.

-v

verbose output (use more than once to increase programs verbosity)

-W file

defines the input coastline file. Defaults to \$APS_DATA/world_01.dat.

-W file=file,draw=[0|1],color=R:G:B,tolerance=t

Sets up options used to draw coastline, where:

keyword	description
draw	draw (1) or not draw (0) the coastline
color	set the color of the coastline
file	define the input file of the coastline
tolerance	defines the tolerance to reduce coastline resolution

The *tolerance* is used by the Douglas-Peucker algorithm to reduce the resolution of the input coastline file. This is useful when the input file and output image contain a large region of the earth, by reducing the number of points draw by the graphics library. A value of 0.05 is good for a coarse coastline for the entire world. A smaller value will yield better resolution.

The *file* is used to set the input coastline file. It can be set to an NSIPS derived coastline file.

-w file

Use file as watermark image. The image will be scaled to fit the product image and overlayed on top of it. This should be an image created with transparency so that when it is overlayed it will give the illusion of a watermark (recommend creating a white image with alpha channel set to 20-30% opacity).

-x

Prevents the drawing of masks on the image, unless dealing with a true color image, then enables drawing of masks on the image. This option will not override masks set with -M, but this option must be used when dealing with true color images in order for the masks set with -M to be drawn.

-X

Prevents the drawing of the coastline on the image.

-y

Overlay true color image based on mask(-z option). If used more than once (as in -y -y) will overlay true color image even for pixels with the invalid value. This is useful so that a static true color image can be created and used for the landmask or so that the land that falls outside of the image coverage will be drawn black instead of given the land mask color(currently brown).

-Y name

Name of product containing true color image. (default: true_color) The product may be in another file. In that case this option is used like -Y filename:prodname.

-Z file

Read in *file* to set options

--help

Print out a small help guide.

--version

Print out version of software and quit.

Files

\$APS_DATA/logo.jpg

This is the default logo file which contains the default logo to apply to the image.

\$APS_DATA/world.dat

This is the default coastline file. It is part of the Naval Satellite Image Processing System (NSIPS).

\$APS_DATA/landmask.dat

This is the default landmask file. It is part of the SeaWiFS Data Analysis System (SeaDAS).

symbols.dat

This file allows the user to define and apply symbols to the image. The format is white space delimited UNIX text file in columnar format. The first two columns are the latitude and longitude in decimal degrees with negative being South or West. The next column contains the symbol type: 0 for filled box, 1 for outline of box, 2 for big filled box, 3 for big unfilled box, 5 for a star). The next column is a scale parameter to increase/decrease the size of the symbol. The next three columns describe the color as an RGB triplet given in floating point values (i.e, 0.0 to 1.0). Starting at text column 42, the rest of the line is used as a text string which will be printed next to the symbol. A '#' as the first character of a line indicates a comment and the line will be skipped.

track.dat

The track file allows the user to define and apply ship track or other transect over the image. The file format is a white space delimited UNIX text file in columnar format. The only two columns are the latitude and longitude in decimal degrees with negative being South or West. A '#' as the first character of a line indicates a comment and the line will be skipped.

blotch.dat

The blotch file allows the user to define and apply polygon areas over the image. The file format is a UNIX text file which begins with a single line containing the number of blotches (n) in the file. This is followed by n groups of lines for which the first line contains the name of the region and the second line contains the number of longitude, latitude (m) pairs in the polygon. The next m lines contain two space delimited columns containing the longitude and latitude of the points on the polygon. All polygons are assumed to be closed. A '#' as the first character of a line indicates a comment and the line will be skipped.

vector files

The vector overlay function can handle files produced by the PC-TIDES, NOGAPS, NCOM, and SeaWiFS/MODIS climatology MET files.

contour files

A contour file must contain the following attributes....

Environment Variables

\$APS_DATA

This environmental variable should point to the APS's data directory. It is used to find the default maps file. If not set, the -M option can be used to specify the user's map file.

Examples

To create a true color image with coastline overlays and other proper annotation from the file S20000001175612.L3_HNAV.

Example 3. Creating a TIFF true color image

```
% setenv APS_DATA /home/aps/aps_v3.4/data
% imgBrowse S20000001175612.L3_HNAV true_color S20000001175612_true_color.tiff
```

Caveats

These are the known problems/bugs with the software.

The maximum image size is limited to 4096 X 4096. This limit is imposed by the off-screen rendering functions of the Mesa 3-D graphics library. This limit can be increased by changing MAX_WIDTH and MAX_HEIGHT in Mesa/src/config.h and recompiling the Mesa library(in fact we have increased these two values from 1280 X 1024), and then recompiling this program with the new library.

This program can not navigate on images which are not warped (like the Level-2 files). Additionally, the warped files must have been created by the program imgMap(1).

All symbols are drawn in white. There is no option to change these.

Name

imgCoords -- add COARDS compliant attributes

imgCoords

imgCoords [*options*] *ifile* [*dataset dataset...*]

Description

imgCoords will update the products either specified on the command line or read from the prodList file attribute with COARDS compliant attributes. The following attributes will be set: long_name, units, format, scale_factor, add_offset, and __FillValue. Additionally, the dimension names for each product will be set to Latitude and Longitude.

Options

-l

Set dimension scales. This is an HDF feature designed to associate navigation with each dimension. The map coordinates for the first row and left-most vertical column are projected and set as the dimension scale for the respective dimension. The 3rd dimension of a 3-dimensional dataset is set to the index. This is only appropriate or recommended for "square" projections such as Mercator or Equi-Rectangular.

-s

Use HDF Library's definition of scale_factor and add_offset: $\text{unscaled} = \text{scale_factor} * (\text{scaled} - \text{add_offset})$. Default is COARDS definition: $\text{unscaled} = (\text{scaled} * \text{scale_factor}) + \text{add_offset}$

-x name

Set x dimension name. Default: Longitude.

-y name

Set y dimension name. Default: Latitude.

-v

Verbose output

--help

Print out a small help guide.

--version

Print out version of software and quit.

Name

imgConvert -- convert APS products to another output image format.

imgConvert

imgConvert [*options*] *ifile* *basename* *dataset* *dataset*...

Description

By default, the program **imgConvert** will output each data set specified on the command line to its own output file. The output file can be one of: (1) a simple binary file; (2) an ENVI formatted file; or (3) a TIFF image file (with GeoTIFF tags if navigation exists). In some cases, much of the navigation and ancilliary data will not be present in the resulting file as some of these formats are not tuned for this type of information.

For binary files, the image is simply written as native floats. The -I option can be used to output the data set in its original type (usually a 16-bit integer). The output files will be named by appending the name of each data set with the extension `.bin`.

For ENVI output, the result is actually two files. One is a binary dump of the input data converted to floating point. The second file is an ENVI header which describes the data in the binary file. If the input file contains navigation information in the NRL format (see `maps(1)`), then the map projection information is written to the ENVI header. *Note:* The NRL projection software is based on the USGS projection code and contains over 30 different projections. Currently, **imgConvert** will only handle the Mercator map projection. The ENVI format may also be written in multibanded format (see -M). For each single-banded output ENVI file (default), the output file name will consist of the *basename* with the name of each data set product and the extension `.envi`. The ENVI header file will have the extension `.hdr` further appended to it. For a multi-banded ENVI file, the *basename* *will be* the output filename. The ENVI header file in this case will be the *basename* with the extension `.hdr`.

For TIFF files, the image data will be written to a TIFF formatted file. If the input HDF file contains navigation, then GeoTIFF tags will be appended to the TIFF file. The default is to write out the data in floating point format. The -I option may be used to leave the data in its stored format. The output file will be named by appending the name of each data set with the extension `.tiff`.

The user may specify the data sets for output by using regular expressions. These must be put in quotes to protect them from expansion by the shell, however.

Options

-B *isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl*

defines a subsection of the original image for output.

keyword	description
isp	the starting sample number
iep	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-d

Turns on debugging output.

-I

Do not convert data from original format.

-m mapFile:mapName

use the following mapFile and mapName for navigation.

-M

Put all bands into a single file (available only for ENVI files).

-s

Makes GeoTIFF tags standard used-tags. Programs such as Falcon View do not support user-defined tags.

-T j|d

Sets the compression option to: JPEG Compression (j) or default compression (d)

-v

Verbose mode

--help

Print out a small help guide.

--version

Print out version of software and quit.

Examples

To create an ENVI file of remote sensing reflectances from the file S2000001175134.N3_HNAV:

Example 4. Extracting Remote Sensing Reflectance Products into a single ENVI file

```
$ imgConvert -M S2000001175134.L3_HNAV S2000001175134.envi "rrs_*"
$ ls S2000001175134*
S2000001175134.envi          S2000001175134.envi.hdr
```

To create a GeoTIFF file of diffuse attenuation at 512 nm from the file S2000001175134.N3_HNAV:

Example 5. Extracting Diffuse Attenuation into a GeoTIFF file

```
$ imgConvert S2000079181416.L3_HNAV_GOM S2000079181416.tiff K_532
$ ls S2000079181416*
S2000079181416_K_532.tiff
```

Notes

The binary files produced by imgConvert are in native format. If the file is transferred across platforms, the user will have to handle any and all byte swapping. For example, if writing out a binary file on an SGI and reading the image into MATLAB on a PC. Some formats (like TIFF) and software (like APS) handle the platform conversion automatically.

Name

imgConvolve -- compute convolution transformation using a kernel

imgConvolve

imgConvolve [*options*] *ifile ofile* [*product product...*]

Description

imgConvolve is used to perform a transformation that gives each pixel in an image a new value that is a function of the pixels in its immediate neighborhood. The image to be transformed is in 2-D product array in ifile and the resulting transformed image is stored in an product written to ofile with the same name. The transformation array is user defined and stored in the ofile array as “kernel” (*currently user can only use the two predefined kernels and the code does not store the kernel in the output file*). Note that if the row number KROW or column number KCOL of the kernel is even, the pixel to the top and left of center is the default central pixel.

The covolution function is:

$$P'(x,y) = \text{Sum of } [K(i,j) * P(x+i,y+j)]$$

where P is the pixel value at column x and row y, K is the kernel, i ranges from -KROW/2 to KROW/2, and j ranges from -KCOL/2 to KCOL/2. The results are not normalized (see -n option, however).

Options

-k type

select from a few predefined kernels

```
1 = Low Pass Filter, 3x3 kernel
  | 1/9, 1/9, 1/9 |
  | 1/9, 1/9, 1/9 |
  | 1/9, 1/9, 1/9 |
2 = Centre-Weighted Edge Detection, 3x3 kernel
  | -1/8, -1/8, -1/8 |
  | -1/8, 1, -1/8 |
  | -1/8, -1/8, -1/8 |
3 = Gaussian Smoothing, 5x5 kernel
  | 1/106, 1/106, 1/106, 1/106, 1/106 |
  | 1/106, 9/106, 9/106, 9/106, 1/106 |
  | 1/106, 9/106, 18/106, 9/106, 1/106 |
  | 1/106, 9/106, 9/106, 9/106, 1/106 |
  | 1/106, 1/106, 1/106, 1/106, 1/106 |
```

-n N

Normalize the array, that is divide P'(x,y) by N.

-S

Sobel edge detection.

-v	Forces imgConvolve to run in verbose mode.
--help	Print out a small help guide.
--version	Print out version of software and quit.

Examples

This call computes a Low Pass Filter on all products in the given file.

Example 6. Applying Low Pass Filter to All Products in a File

```
$imgConvolve -k1 MODPM2004011194000.L3_NOAA_MSB junk
```

This call computes the Center-Weighted Edge Detection on only the remote sensing reflectance products.

Example 7. Applying Center-Weighted Edge Detection to only Reflectance Products in a File

```
$imgConvolve -k2 MODPM2004011194000.L3_NOAA_MSB junk2 'rrs_*
```

Name

imgDiff -- compute difference between product(s) in two files.

imgDiff

imgDiff [*options*] *ifil1 ifil2 ofile* [*product product...*]

Description

This program computes a simple difference for all products in two input files writing the result to a third file. The user may select the products, which must exist in both input files (though the -A option will allow of alias names to be created), on the command line after the output filename. If no products are given on the command line, then imgDiff will use the file attribute 'prodList to obtain the list of products if it exists.

If the output file is actually one of the input files, then the product name for the difference image will be the product name plus the term "_diff".

The simple difference is computed as $I_1 - I_2$. The difference is only calculated for good pixels which are those that have not been masked (e.g., set to LAND/CLDICE) and are not invalid. If the pixel was masked out the result will be zero. If the pixel is invalid in either image, it is considered invalid in the output. The difference is normally written to the output file using the same scaling as the input file. This can be overridden using the -f option

Besides the simple difference, imgDiff can compute the percent change, percent difference, and the ratio of the two input images. These products will be added to the simple difference (unless the -d option is used). The names of these products will be the product name plus "_per_chg", "_per_diff", and "_ratio".

The percent change is defined as: $I_1 - I_2 / I_2$

The percent difference is defined as:

$$\frac{I_1 - I_2}{\frac{I_1 + I_2}{2}}$$

The above two products are normally written out as unsigned 16-bit integers with a resolution of 0.01 per scale. Thus, the output contains value in the range from -327.67% to 327.67%. If more precision is required, the -f option will output these as floating point images.

The ratio is defined as I_1 / I_2 . If either value is zero, the ratio will be defined as one.

Options

-A name=alias,...

This option allows the user to give alias's to products so that differently named products may be differenced. For example, one file may contain chl_oc3 and the other chl_oc3m. Thus the option -A chl_oc3=chl_oc3m will then match the chl_oc3 product from the first file with the chl_oc3m product from the second file.

-c

Additionally compute the percent change.

-D

Only write output file if a difference is actually computed. If the two files show no difference this option will cause not output to be generated.

-d

Turn on debugging messages.

-f

For each product, the difference will be written using the same data calibration used as the input file product. This option allows the user to force the differences to be written as floating point.

-k

Count up the number of images that have differences and return this number as an exit status.

-m min,max

Use the given range for data filtering when computing the difference. The range is applied to both data sets.

-n

Do not write output file. If this option is used, then the command line should not contain the output file name. Normally used with the -k and -v options.

-o name=name,format=format,conv=conv

Define output file.

keyword	description
name	name of the output file
format	format of the output file
conv	convention of the output file

-p

Additionally compute the percent difference.

-q

Silence all output.

-r

Additionally compute the ratio.

-R

Perform a raw data file comparison. In this case, the two products are compared uncalibrated. For example, if a product is defined as `int16` types in both files, the actual integers are compared. This option precludes any of the other difference products (difference, percent change, and ratio).

-s slope,int

This option allows the user to specify the scaling slope and intercept to use for the output product. By default, the scaling of the product in the first file is used. Care must be taken when using this option as it applies to ALL output products.

-t 0|1|2|3|4|5

This sets the output type and is used in combination with the -s option. The values are: (0) for same as type/scaling as input, (1) for 8-bit integer, (2) for 16-bit integer, (3) for 32-bit integer, (4) for floating point, and (5) for double precision.

-x

Turn off masking check.

-v

Forces imgDiff to run in verbose mode.

--help

Print out a small help guide.

--version

Print out version of software and quit.

Examples

In the first example, all the products in the 2.3 file (stored in the file attribute 'prodList') will be subtracted from the 2.4 file with the differences stored in the DIFF file. The product names will be consistent across all files. That is, the product 'rrs_412' found in both the 2.3 and 2.4 files, will yield a product called 'rrs_412' in the DIFF file.

Example 8. Producing a Difference Between Files Created Using Two Versions of APS

```
$imgDiff S2000208182716.L3_HNAV_2.4 S2000208182716.L3_HNAV_2.3 \  
        S2000208182716.L3_HNAV_DIFF
```

In this case, we only compute the difference images for the remote sensing reflectances.

Example 9. Producing a Difference Between Files Created Using Two Versions of APS For Only Reflectance Products

```
$imgDiff S2000208182716.L3_HNAV_2.4 S2000208182717.N3_HNAV_2.3 \  
        S2000208182716.L3_HNAV_DIFF "rrs_*
```


Below is an example of using the same output file as input file. Note, that in this case, the output products will be stored back in the 2.4 file with the names: "rrs_412_diff", "rrs_443_diff", etc.

Example 10. Producing a Difference Between Files Created Using Two Versions of APS For Only Reflectance Products (Output Same as an Input)

```
$imgDiff S2000208182716.L3_HNAV_2.4 S2000208182717.N3_HNAV_2.3 \  
        S2000208182716.L3_HNAV_2.4 "rrs_*"
```

If the user is only interested in whether there are differences and not in the differences themselves, then the -k option will be useful. Thus it falls that we are normally not interested in the output file, so the -n option is normally selected. Here is an example, using a shell script:

Example 11. Determine Difference Results on a Series of Input Files

```
dir1=/rs/lvl3/seawifs/2.3/MissBight/2001/jan  
dir2=/rs/lvl3/seawifs/2.4/MissBight/2001/jan  
find $dir1 -type f > /tmp/a.list  
find $dir2 -type f > /tmp/b.list  
list=`cat /tmp/a.list /tmp/b.list | sort | uniq`  
for f in $list  
do  
    if imgDiff -kn $dir1/$f $dir2/$f K_532  
    then  
        echo $f differs  
    fi  
done
```

Name

imgDump -- dump data from HDF file

imgDump

imgDump [*options*] *ifill ofile prod*

Description

This program is used to make ASCII dumps of data from an HDF file. By default, each pixel over land or with a mask value of non-zero (using the default mask of LAND, CLDICE) will be printed. The user may select from one of several desired output formats (see *FORMATS*).

Options

-B isp=isp,iel=iel,isl=isl,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iep	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

Selects the desired format. May be either *fox* (default) *ko* or *latlon*.

-F name

Name of input mask data set (default *l2_flags*)

-L file

Use give file for the land mask file (default *\$APS_DATA/landmask.dat*)

-M names

A comma separated list of flag names to use for data masking (default is *LAND,CLDICE*).

-r m,n

This filters the data by range. Only data falling within these limits will be dumped.

-v

Verbose output.

--help

Print out a small help guide.

--version

Print out version of software and quit.

Formats

The following ASCII formats are available.

FOX

It was originally written to be used as input to Dan Fox's model and therefore output's the Dan Fox File.

**152 YYYY MM NNNNN.NN
EEEE.EE DDD HHH SSS.SS**
YYYY4-digit year
MM2-digit month
NNNNN.NN8.2 latitude
(degrees North)
EEEE.EE8.2 longitude
(degrees East)
DDD3-digit day of
month
HHH3-digit hour of
month
SSS.SS5-digit data value
(*e.g.*, sea surface
temperature)

KO

It was originally written to be used as input to Ko's model.

**YYYY MM DD HH MM SS
NNNNN.NN EEEEE.EE DDD
HHH SSS.SS**
YYYY4-digit year
MM2-digit month
DD2-digit day of
month
HH2-digit hour of
day
MM2-digit minute of
day
SS2-digit second of
day
NNNNN.NN8.2 latitude
(degrees North)
EEEE.EE8.2 longitude
(degrees East)
SSS.SS5-digit data value
(*e.g.*, sea surface
temperature)

LATLON

This is a simple format that includes latitude and longitude and data.

NNNNN.NNN EEEEE.EEE

SSSSSS.SSSSS

NNNNN.NNN9.3 latitude
(degrees North)

EEEE.EEE9.3 longitude
(degrees East)

SSSS.SSSSS11.5 digit data
value (*e.g.*, sea
surface
temperature)

Name

imgFillGaps -- fill in data gaps

```
imgFillGaps  
imgFillGaps [options] ifile [product product... ]
```

Description

imgFillGaps will....

Options

-i	enable iteration.
-j int	limit number of iterations.
-l	disable use of l2flags.
-L file	Use file for landmask.
-I name	mask string for ignore pixels
-M name	mask string for replace pixels
-m int	set maximum search distance
-n	enable nearest neighbor fill
-v	Verbose output
--help	Print out a small help guide.
--version	Print out version of software and quit.

Name

imgLandMask -- create a land mask product

imgLandMask
imgLandMask [*options*] *ifile* [*ofile*]

Description

This program is used to create a landmask product for the given file. The resulting landmask will be written as 2-D (or 3-D) byte data set with land pixels represented by the value 255 and the water pixels represented by the value 0. If the land and water pixels are given as RGB triplets, then the output data set will have three dimensions. The output data set will be named land_mask.

By default, the input landmask file is \$APS_DATA/landmask.dat file.

Options

-B isp=isp,iep=iep,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iep	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-l # -or- **-l** r,g,b

Output land pixels using the given value, which must be between 0 and 255.

Note

Land and water pixels must have separate values.
The second option will set the land pixels to the given RGB triplet and produce a 3-D data set.

-L landmask file

This option is used to specify the input landmask file.

-n name

This option is used to specify another name for the output data set. The default is land_mask

-w # -or- **-w** r,g,b

Output 'water' pixels using the given value, which must be between 0 and 255.

Note

Land and water pixels must have separate values.
The second option will set the land pixels to the given RGB triplet and produce a 3-D data set.

--help

Print out a small help guide.

--version

Print out version of software and quit.

Environment Variables

APS_DATA

The directory where all the data files exist. Defaults to \$APS_DIR/data.

Files

landmask.dat

The landmask file.

Name

imgMakeLatLon -- create "latitudes" and "longitudes" products for image

```
imgMakeLatLon
imgMakeLatLon [options] ifile [ofile]
```

Description

This program is used to create a latitude/longitude products for the given input file. The resulting products are stored as 32-bit floating point numbers as decimal degrees (- west, + east). They are the same size as the arrays in the input file and are named latitudes and longitudes by default. This program is useful when latitude/longitude is needed for every point in the image.

Options

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-l name

Rename the latitudes data set to name.

-L name

Rename the longitudes data set to name.

-x

This option will write the map coordinates (in meters) to the file. The products will be named mapX and mapY.

--help

Print out a small help guide.

--version

Print out version of software and quit.

Name

imgMap -- project satellite images to map projection

imgMap

imgMap [*options*] *mapName ifile ofile [product product...]*

Description

This program is used to project navigatable APS products to a map projection specified by the user. The user should use the program maps(1) to create an “image map” that is, an image with a defined number of samples and lines and projection system. Each image map is usually stored in a single file called “maps.hdf” This file contains a series of user-defined image maps given a unique name.

The input file must contain either a latitude/longitude product (for each pixel) in the data file or a control points grid. The control point grid is usually defined with the products “CP_Pixels” “CP_Lines” “CP_Latitudes” and “CP_Longitudes” These are created by default by various programs within APS. Using these points to navigate over the input image (usually in the satellite sensor projection), **imgMap** will fill in the output array from the nearest pixel in the input image. This program does not perform any type of interpolation.

The list of products can use regular expressions. The user should quote them, however, so that they are not interpreted by the UNIX shell.

This program can handle 2-D and 3-D input arrays. It is assumed that the 3-D data is stored in BIP format.

The program will append the folling file attribute to the output data file: mapProjection. This attribute will point to the product which contains the mapName. The mapName product will automatically be appended to the output file.

Options

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that

number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-d

Turn on debugging messages.

-I

Define the file (when producing mosaics using *-m*) which is appended to the inputFiles attribute.

-m

Set the file type parameter to be a mosaic file.

-M mapFile

Use the given mapFile rather than the default version.

-v

Forces imgMap to run in verbose mode.

--help

Print out a small help guide.

--version

Print out version of software and quit.

Files

`$APS_DATA/maps.hdf`

This is the default mapFile which contains the map provided on the command line.

Environment Variables

`$APS_DATA`

This environmental variable should point to the APS's data directory. It is used to find the default maps file. If not set, the *-M* option can be used to specify the user's map file.

Examples

This example will use warp all the remote sensing reflectance images located in S20000001175612.L2_HNAV file using the MissBight image map located in the file ~/ladner/maps.hdf.

Example 12. Warping Reflectance Data Using Specified Map

```
% imgMap -M /home/ladner/maps.hdf MissBight \  
S20000001175612.L2_HNAV S20000001175612.L3_HNAV "rrs_*
```

If \$APS_DATA is set then, then user does not have to use the -M option.

Example 13. Warping All Data Using Specified Map

```
$ export APS_DATA=/home/aps/aps_v3.4/data  
$ imgMap GulfOfMexico S20000001175612.L2_HNAV S20000001175612.L3_HNAV
```

See Also

maps(1)

Name

imgMean -- calculate Mean/Min/Max/StDev of a series of images.

imgMean

imgMean [*options*] *ifil1 ifil2...*

Description

This program will produce an image of the mean for each pixel in a series of images. The program can also produce an image of the minimum, maximum, standard deviation. Currently the program is limited by the HDF libraries to composites of at most 31 files. A shell script has been created that when used with the -F option can work around this limit.

In addition, this program can make incremental composites (the addition of one or more files to an already created composite file) and can merge two or more composite files. Names of composite files may be placed on the command line or in an input file(when using -F) just like regular Level-3 files, they will be recognized as NRL Level-4 files.

For *l2_flags* products, the compositing is done as follows. Until we find a compositable pixel, *l2_flags* are OR'ed. should no compositable pixels be encountered, the resulting *l2_flag* is the OR of all of the *l2_flags* of the images.

Once a compositable pixel has been found, the *l2_flags* are AND'ed with other compositable pixels.

Options

-a	Do not create the average (mean) image.
-c	Output a "count" product.
-C product	The name of the product containing the cloud albedo for each pixel. Default: "cloud_albedo".
-F filename	Get files to composite from "filename". The format of this file is one file name per line, the whole line is used, and may contain spaces or any other character. DO NOT QUOTE FILENAMES. Any combination of Level-3 and NRL Level-4 files may be used.

-f #

Apply the given function to the input data before determining any statistics with the data. The number represents the available functions which are: 0 for none, 1 for log10, 2 for alog10, 3 for ln, 4 for exp. The function is applied after any range checks are performed. Multiple functions may be specified by separating the functions by a comma. If fewer functions are specified than products, no function will be used with the remaining products. The -f option is only valid when -H is also specified.

ex. -H K_532,ch1_oc4 -f 0,1

K_532 function none ch1_oc4 function log10

or

-H K_532,ch1_oc4,bb_555_arnone -f 0,1

K_532 function none ch1_oc4 function log10 bb_555_arnone function none

-H product

Designates that the files to be used in making composites will be in HDF format. The user must supply the name of which product to use. Multiple product's may be specified by separating them with comments.

ex. -H K_532,ch1_oc4

-I #

Used to specify the value to be used in replacing invalid data.

-l

Composite the l2_flags.

-L

Do an incremental composite.

-m #

The mask value used to filter out pixels from the compositing. The mask may be specified as an integer or as a comma separated string of flag names.

ex. To mask out the ATMFAIL and LAND flags, use either of the following: -m 3 or -m ATMFAIL,LAND

-M product

The name of the mask array to use for masking. The default is "l2_flags".

-n

Produces an image of the minimum value of all input images at each corresponding pixel location.

-o outfile

Set the output filename to "outfile". If this option is not used then an output filename will be created based on the start and end times of the input files. The name will be of the form `YYYYYDDDDYYYYYDDDD.L4_TT_REGION` where the first group of `YYYYYDDDD` is the earliest start year and julian day and the second group is the latest end year and julian day, the `TT` is the composite type as set by the `-T` option. `TT` will be `WE` for weekly, `MO` for monthly, `YR` for yearly, and `RO` for Rolling composites. If a daily composite was specified then the name will be `YYYYYDDDD.L4_REGION`. `REGION` is a short string describing the region. (e.g. `GOM` for `GulfOfMexico`)

-p #

Define the number of valid pixels to use. Default value is 1.

-P d,i

Produces an image of the latency of all input images at each corresponding pixel location. This product replaces the count image in latest pixel composites and forces a maximum valid pixel value of 1 (*i.e.*, `-p 1`). When the `-P i` option is used, the product will contain the file index (1-relative) of the file used in the latest pixel composite. The file used can be found in the *inputFiles* file attribute. For pixels that are invalid, a index of zero will be written. When the `-P d` option is used, the pixel will contain the number of days from the most recent file. Thus, a value of 0 will indicate pixels that are the most recent. A value of 7 will indicate pixels that are seven days old. Invalid pixels will contain `USHRT_MAX`.

-r #,#

Set the lower and upper bounds for range checking. A lower and upper bounds may be specified and must be separated by a comma. Ranges for multiple products may be specified by separating the ranges by a colon. The `-r` option is only valid when the `-H` option is also specified.

ex. `-H K_532 -r .01,6` or `-H K_532,chl_oc4 -r .01,6:.01,64`

If fewer ranges are specified using `-r` than products using `-H` then the last value in the `-r` list will be for the rest of the `-H` products.

ex. `-H K_532,chl_oc4, chl_stumpf,... -r .1,5:.1,64`

`K_532` range .1 - 5 `chl_oc4` range .1 - 64 `chl_stumpf` range .1 - 64 ... range .1 - 64

If no ranges are specified, then the ranges from the 'validRange' attribute are used for range checking.

-s

Produces an image of the standard deviation for each corresponding pixel for all input images.

-t albedo

This defines the albedo value to use when compositing data. To be used the input files must contain the *cloud_albedo* product.

- T #
Specify type of composite. 1 for daily, 2 for weekly, 3 for monthly, 4 for yearly, 5 for latest pixel composite, 6 for rolling composites and 7 for seasonal. This is mainly an informational option to describe the composite. It adds a suite for attributes including *compType*, *compStartTimeFrame*, *compEndTimeFrame*, *compTimeFrame*.
- v
Forces imgMean to run in verbose mode.
- W weights
Creates a weighted average based upon the weight table given in weights. The output product for the weighted average will be "_weight".

NOTE Currently incremental composites are not possible with this option.

weights is defined as a string of comma "," seperated floating point numbers which define your weight table.

ex: -W .6,.2,.1,.05,.04,.01

would define a weight table as follows

[all valid pixels in the first input file] * .6 [all valid pixels in the second input file] * .2 [all valid pixels in the third input file] * .1 etc.

The sum of the valid pixels * their respective weights is divided by the addition of those weights that were used for each pixel location.

ex. 5 files are input, using a weight table of .5,.4,.3,.2,.1 for pixel location p files 1,2, and 5 have valid data, the weighted mean would be calculated as follows: ([pixel 1] * .5 + [pixel 2] * .4 + [pixel 5] * .1) / (.5 + .4 + .1)
- w start,stop
Define the week start and stop times for the -T2 composites. Used when creating *compStartTimeFrame/compEndTimeFrame* attributes.
- x
Produces an image of the maximum value of all input images at each corresponding pixel location.
- z #
Set the sensor zenith angle threshold. imgMean will ignore pixels whose sensor zenith angle(degrees) is above this threshold.
- Z product
The name of the product containing the sensor zenith angles(degrees) for each pixel. Default: "senz".
- help
Print out a small help guide.
- version
Print out version of software and quit.

Examples

This example will composite together all sea surface temperature values for NOAA-14 for day 29 of 2002.

Example 14.

```
$ imgMean -H sst day029.hdf /rs/lvl3/avhrr/4.0/GulfOfMexico/2002/jan/ND2002029*
$ hdf day029.hdf list
** FIXME **
$
```

The file name given to the file is in the attribute "file". The "fileTitle" attribute has been set to "NRL Level-4 Data". "inputParameters", "inputMasks", and "inputMasksInt" contain their respective values. The "timeStart*" and "timeEnd*" attributes contain the earliest start times and latest end times respectively of the input files. "inputFiles" contains a list of the files used to create the products. Also, notice that the mean product is defined by appending the product with "_mean", as would be true for the std. dev. ("_dev"), max ("_max") and min ("_min").

Name

imgRGB -- Create RGB from 3 products.

imgRGB

imgRGB [*options*] *ifile ofile redprod greenprod blueprod*

Description

This program is used to create a true color (rgb) image from three products. The output file will contain a 3 banded data set named by default "true_color". The output file name may be the same as the input file name.

Options

-b switchpoint,high_slope,high_offset,low_slope,low_offset

These values are used for *-s bilinear* scaling and determine the slope and offsets for each segment and where the switch point (or knee) is located. Defaults are switchpoint = 0.2, high_slope = 1.2, high_offset = 0.0, low_slope = 1.9, low_offset = 0.0.

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension(not implemented)
irl	the replication factor along the lines dimension(not implemented)

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output. Cannot be used with the -s option.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-p name

Use name as RGB product name. Default is *true_color*.

-N long_name

Provide the long name of the product.

-r r1,r2

Set range of input data, defaults to 1,100 for all bands. Ranges for the three bands may be set individually by separating each range by a colon.

ex. -r .01,1:.02,.9:.08,.6

-R r1,r2

Set range of output image. Defaults to 0,255.

-s type

Set output scaling type. A number of scaling options are available including linear, log, bilinear, 2nd degree polynomial and 3rd degree polynomial scaling. For bilinear scaling the -b option is used. The 2nd degree polynomial equation is $c = -1.5686 b^2 + 2.5686 b$, where b is the second band (red, green, or blue) and c the scaled image channel (0-255). The 3rd degree polynomial is $c = -2.8115 b_3 + 2.605 b_2 + 1.2065 b$.

-t	Define each channel to be the percentage of that channel to the total (or sum) of all channels. That is $c(i) = c(i) / [r(i) + g(i) + b(i)]$.
-v	Increase verbosity.
--help	Print out a small help guide.
--version	Print out version of software and quit.

Examples

The following examples creates the *water_mass* product using the portions of the total absorption.

Example 15. Producing a Water Mass Classification Product from Selected Input Data

```
$imgRGB -N "Water Mass Classification" -p water_mass -t_  
S2005016181757.L3_HNAV_MSB  
water_mass.hdf ad_412_gould aph_443_gould acdom_412_gould
```

Name

imgRead -- dump information from images

imgRead

imgRead [*options*] *ifil product product...*

Description

The program imgRead allows the user to retrieve data from an image at any desired position specified either by (latitude,longitude) pair or (line,sample) pair. The values are read as geophysical values and dumped to stdout. The user may select a single point or a square around the specified position. Eight different box sizes specify an area from 3x3 to 17x17 pixels.

Caveat

The latitude/longitude options can be used only with map projected files and must use the -m option. All others (Level-1, Level-2, etc.) can only use line,sample option, i.e. -x option.

Options

-b size

Use a box around the point of interest. Should be one of 3, 5, 7, 9, 11, 13, 15, 17.

-c

Output data in columns. If used with the -b option, this will output the average of the box in the column. Cannot be used with the -S option.

-f %#.##f

Used to control output formatting with the -b option. The number is the number of spaces for the entire number and the second number represents the number of decimal places. For example: -f %10.5f will output: xxxx.xxxxx Default format is %10.4f.

-F flagName

Show the flag specified as *flagName* as a 16 bit binary number. The output will be a string of 16 one's or zeros.

-g outputFile

Used to output data in a format acceptable by the GNU program graph. The argument outputFile should be the name of the output file to be created. The name of the product extracted will be appended to the outputFile name.

-h

This option is used to suppress the headers.

-m mapFile:mapName

This option is used when a mapped file will be read by the program. The first string should be that of the maps file followed by a colon and the name of the map. For example, -m maps.hdf:ChesapeakeBay.

-n name	Provides the name of the station which will be included in the header provided the -s option is used.
-p [x y]	This option outputs a row (y) or a column (x) profile of the data.
-r min,max	Set the minimum and maximum range for the data used to calculate the output statistics when using the box option.
-s	Set to include the station name given by -n option in the output header.
-S	This will force the output to look similar to the Seadas output. Cannot be used with the -c option.
-t	Insert tabs between columns when using columnar output.
-x	Treat input values as samples and lines.
--help	Print out a small help guide.
--version	Print out version of software and quit.

Examples

To dump a series of points read in from file stations.dat and output to data.dat.

Example 16. Extracting Data From a Product File

```
$ more stations.dat
37.4502 -89.3403
37.5320 -89.3403
-99.0 -99.0
$ imgRead S1998100175129.N3_HNAV_MSB nLw_412 nLw_443 < stations.dat > data.dat
$ more data.dat
Latitude Longitude Pixel Line nLw_412 nLw_443
37.4502 -89.3403 302 142 -0.2040 0.0300
37.5320 -89.3403 303 141 -0.0010 0.2160
```

Name

imgReformat -- tile and compress an HDF SDS.

imgReformat

imgReformat [*options*] *ifile ofile xnumchunks ynumchunks sds1 sds2...*

Description

This program is used to convert an unchunked SDS into a chunked and compressed SDS. The number of chunks across the array are given on the command line as xnumchunks and ynumchunks. Each chunk will be compressed using the gzip deflate compression scheme at a level of 6 by default. To specify an encryption method the user may use command line options as defined below. Three dimensioned chunks are supported but they will be chunked along the XY with Z being constant to it size.

Ex: imgReformat -t 4 -l 7 outFile 5 5 sds1 sds2 sds3 Would use level 7 gzip compression on 5x5 chunks of sds(1-3) with the output being placed in outFile.

Options

--help

Print out a small help guide.

--version

Print out version of software and quit.

-f num

Optional format 1=BIP 2=BII 3=BSQ

-t num

Optional compression method 1 = RLE 2 = NBIT (not supported) 3 = Skipping Huffman 4 = GZIP (default)

-l num

Deflation level for GZIP algorithm [0-9]

-s num

Skip size for Skipping Huffman algorithm [>1]

See Also

HDF User Guide

Name

imgSmooth -- apply filter, perform statistics

```
imgSmooth  
imgSmooth [options] ifile ofile [product product... ]
```

Description

This program is used to perform a transformation that gives each pixel in an image a new value that is a function of the pixels in its immediate neighborhood. The replacement function may be either the mean or the median.

In addition, the standard deviation of the pixels used to calculate the new pixel value may be calculated, along with the maximum and minimum pixel value. The function is applied to the input image by sliding a window over the image and applying the function to the pixels that fall underneath the window, in order to replace the center pixel in the window with the newly calculated value. If the window dimensions are even, then the center pixel is the pixel to the top and left of center. Invalid pixels and pixels flagged by a `l2_flags` mask are ignored during processing. The output file name may be the same as the input file name.

Options

- a
- Aggregate, slide window across image so that the pixels covered by the window used to create adjacent pixels does not overlap. This will create a smaller image.
- For example:
- a 10 x 10 product with a 2 x 2 mask size will produce a 5 x 5 output product.
 - an 11 x 11 product will produce a 6 x 6 output product
- This option cannot be used with `iterate`, in addition the output file name must be different than the input file name since the product dimensions will change.
- d num
- Set maximum number of iterations. Note, iterations start after the first pass over the data.
- i num
- Iterate, continue smoothing until standard deviation falls below "num" threshold.
- j num
- Set `l2_flags` mask. Default: 523 = ATMFAIL, LAND, HIGLINT, CLDICE.
- k
- When iterating, compare average standard deviation of entire image to threshold. Default: compare greatest standard deviation of entire image.
- l
- Use `l2_flags` if available.
- m num
- Use num X num window.

-M	Do median filter. Default: mean filter.
-n	Create minimum neighbor image.
-q	Use quick sort when performing median filter. Default: radix sort. Quick sort may be faster than radix sort for small mask sizes such as 2 x 2 or 3 x 3.
-S	Create standard deviation of neighborhood image.
-v	Increase verbosity.
-x	Create maximum neighbor image.
--help	Print out a small help guide.
--version	Print out version of software and quit.

Examples

This smooths only the rrs_412 image

Name

imgStat -- calculate statistics and data quality products of an image, and creates histograms.

```
imgStat
imgStat [options] ifile [product product... ]
```

Description

imgStat is used compute statistics and quality of data. It can generate histograms, a quality product - based on l2_flags, and quality attributes.

Options

-B isp=isp,iép=iép,isl=isl,iel=iel,irp=irp,irl=irl

Define a subsection of the original image for output.

keyword	description
isp	the starting sample number
iép	the ending sample number
isl	the starting line number
iel	the ending line number
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat. If set to a negative number each sample is repeated the number of times equal to the absolute value of that number. Thus a positive irp is used to reduce or shrink the image and a negative irp is used to enlarge or magnify the image.

-B nlat=nlat,slat=slat,wlon=wlon,elon=elon,irp=irp,irl=irl

Define a subsection of the original image based on geographical coordinates for output.

keyword	description
nlat	the latitude of most North Western point
slat	the latitude of most South Eastern point
wlon	the longitude of most North Western point
elon	the longitude of most South Eastern point
irp	the replication factor along the samples dimension
irl	the replication factor along the lines dimension

The irp/irl indicates the number of samples/lines to skip or repeat (see previous -B description).

-f function

Sets the scaling to one of the following:

keyword	description
lin	linear
log	logarithm on x-axis
loglog	logarithm on x and y-axes

-F name

Name of input mask product , defaults to *l2_flags*

-n nbins

Defines the number of bins to use; defaults to 100.

-r min,max

Sets the defined range of data for the histogram. If not set, this will be obtained from the data file. First by looking of the attribute *validRange* and then to *browseRanges*.

-v

Verbose output

--help

Print out a small help guide.

--version

Version information

Examples

This example will calculate statistics on the chl_oc4 product in the hdf file

Example 17. Computing Statistics For a Product

```
$ imgStat S2004153172616.L3_HNSG_GOM chl_oc4
1057188 DBL_MINs set
size 4398300, total 1252084
xmin = 0, xmax = 50, xtick = 10
ymin = 0, ymax = 1e+06, ytick = 100000
delta = 0.4499, min 0.01
 0 0.01) 907141
 1 0.46) 222694
 2 0.91) 42679
 3 1.36) 17046
 4 1.81) 12847
 5 2.26) 10405
 6 2.71) 7322
 7 3.16) 5509
 8 3.61) 4569
 9 4.06) 4358
10 4.51) 3640
11 4.96) 2749
12 5.41) 2176
13 5.86) 1670
14 6.31) 1353
15 6.76) 916
16 7.21) 696
17 7.66) 522
18 8.11) 433
19 8.56) 361
20 9.01) 295
21 9.46) 275
22 9.91) 228
23 10.36) 174
24 10.81) 168
25 11.26) 144
26 11.71) 75
27 12.16) 114
28 12.61) 60
29 13.06) 94
30 13.51) 54
31 13.96) 69
32 14.41) 54
33 14.86) 18
34 15.31) 32
35 15.76) 60
36 16.21) 22
37 16.66) 11
38 17.11) 26
39 17.56) 29
40 18.01) 36
41 18.46) 14
42 18.91) 60
43 19.36) 27
44 19.81) 5
45 20.26) 11
46 20.71) 19
47 21.16) 25
48 21.61) 6
49 22.06) 28
50 22.50) 8
51 22.95) 10
52 23.40) 27
```

Name

imgTSeries -- compute difference between product(s) in two files.

```
imgTSeries
imgTSeries [options] ifile roi ofile product
```

Description

imgTSeries is used to perform statistics on a region of interest and output the results in an ASCII file.

Options

-F name	Name of the mask product. Defaults to “l2_flags”
-L name	Name of the input land mask file product. Defaults to \$APS_DATA/landmask.dat
-r min,max	Set the minimum and maximum of the range of data to consider. Defaults to values set in “validRange” attribute.
-v	Verbose output
--help	Print out a small help guide.
--version	Print out version of software and quit.

Name

roi -- Region of Interest

roi

Description

A “blotch” or Region of Interest (ROI) file is a simple ASCII file that contains one or more regions that are defined as closed convex polygons in geographical coordinates. The coordinates are in decimal degrees with ranges from (-180.0 to 180.0) or (-90.0 to 90.0) for longitudes and latitudes, respectively. A positive value is East or North for longitude and latitudes, respectively.

The # is used as comments. The file must start with a line containing the number of blotchs within the file. For each area of interest, the first line contains the name of the region followed by the number of points in the polygon. The next n lines contain the longitude and latitude for each point.

An example of the format is given below:

```
# This file contains 1km resolution blotchs for the
# Gulf of Mexico region
#
# How many?
4
#
# Mobile Bay Polygon
#
Mobile Bay
155
-88.1098 30.3595
-88.1062 30.3748
.....
-88.1098 30.3595
#
# Lake Ponchatrain
#
Lake Ponchatrain
126
-89.7408 30.1737
.....
```